Fatima Jinnah Women University

Opening Portals of Excellence Through Higher Education

“OPEN ENDED LAB”

COURSE:

SOFTWARE CONSTRUCTION AND DEVELOPMENT LAB

SUBMITTED TO:

SIR MUHAMMAD SHAHZAD

SUBMITTED BY:

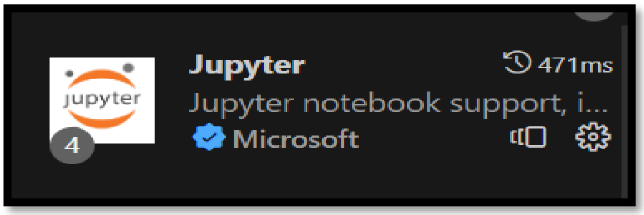
LAIBA SOHAIL (2021-BSE-016)

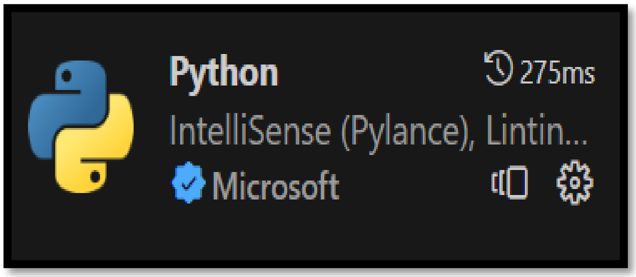
TANZEELA ASHAR (2021-BSE-032)

BISMA ALI (2021-BSE-008)

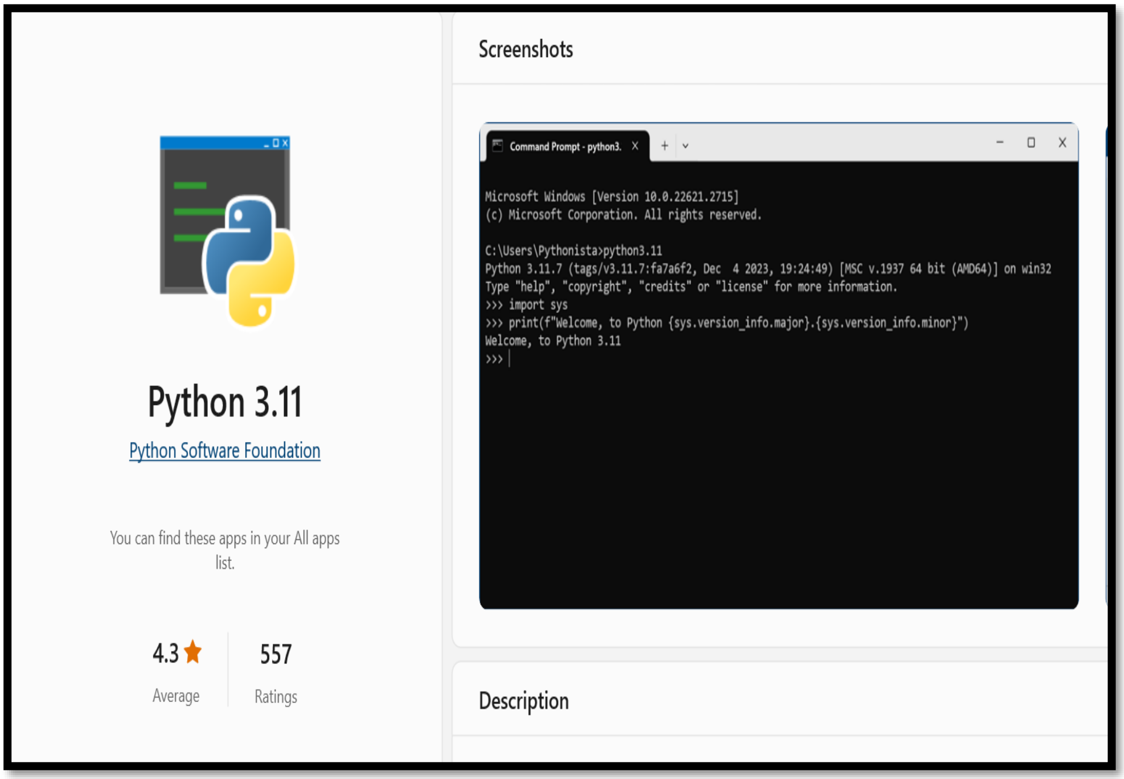
LIVE CRICKET SCORE TRACKER

INSTALLATION OF EXTENSIONS:





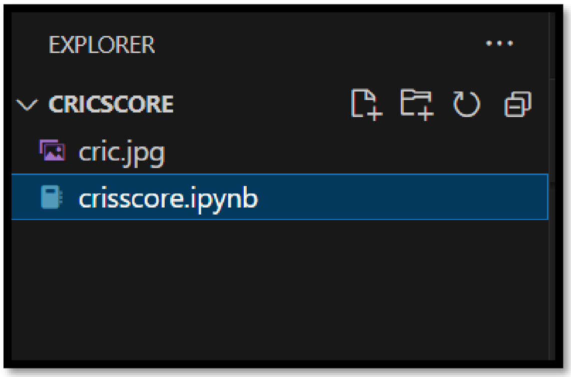
INSTALLATION OF PYTHON:



You can find these apps in your All apps list.

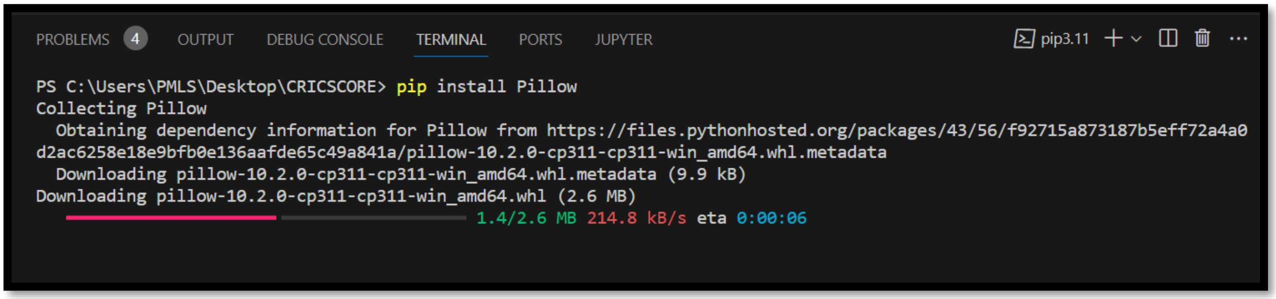
```
Microsoft Windows [Version 10.0.22621.2715]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Pythonista>python3.11
Python 3.11.7 (tags/v3.11.7:fa7a6f2, Dec  4 2023, 19:24:49) [MSC v.1937 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import sys
>>> print(f"Welcome, to Python {sys.version_info.major}.{sys.version_info.minor}")
Welcome, to Python 3.11
>>> 
```

MAKE A FOLDER AND INSERT FILES IN IT:

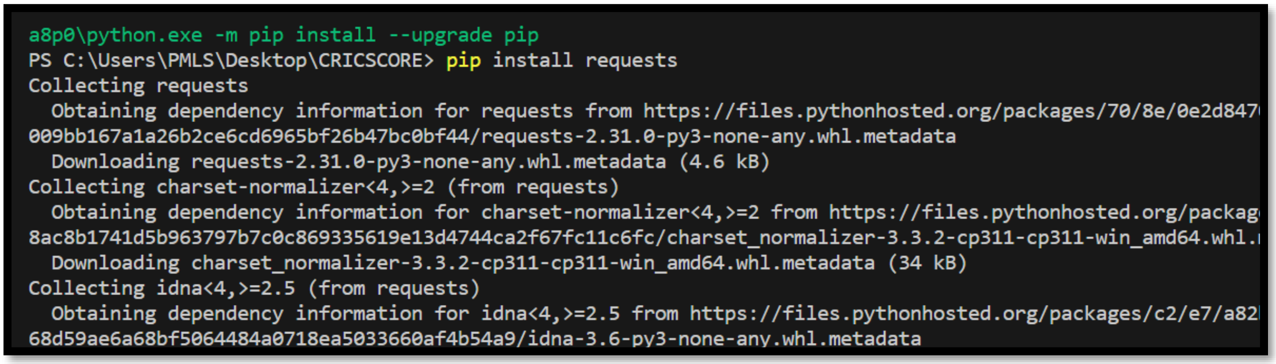


INSTALLATION OF COMMANDS ON TERMINAL:

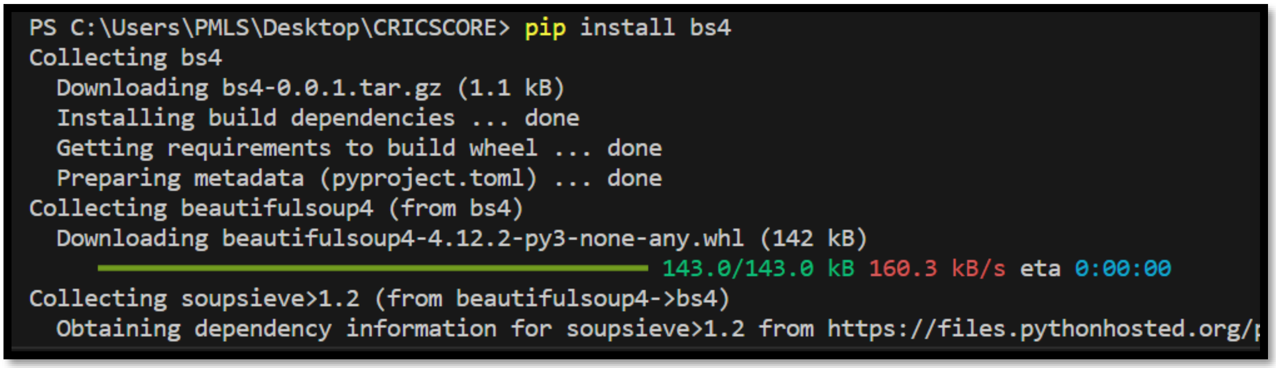
- **pip install Pillow**
- **pip install bs4**
- **pip install requests**



```
PS C:\Users\PMLS\Desktop\CRICSCORE> pip install Pillow
Collecting Pillow
  Obtaining dependency information for Pillow from https://files.pythonhosted.org/packages/43/56/f92715a873187b5eff72a4a0
d2ac6258e18e9bfb0e136aafde65c49a841a/pillow-10.2.0-cp311-cp311-win_amd64.whl.metadata
  Downloading pillow-10.2.0-cp311-cp311-win_amd64.whl.metadata (9.9 kB)
Downloading pillow-10.2.0-cp311-cp311-win_amd64.whl (2.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.4/2.6 MB 214.8 kB/s eta 0:00:06
```

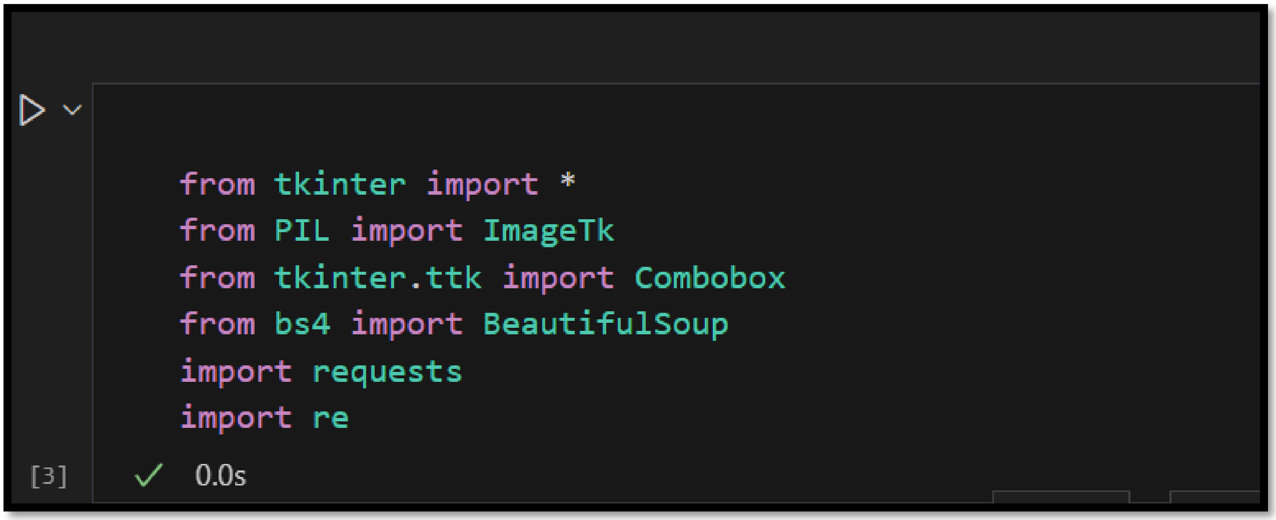


```
a8p0\python.exe -m pip install --upgrade pip
PS C:\Users\PMLS\Desktop\CRICSCORE> pip install requests
Collecting requests
  Obtaining dependency information for requests from https://files.pythonhosted.org/packages/70/8e/0e2d847
009bb167a1a26b2ce6cd6965bf26b47bc0bf44/requests-2.31.0-py3-none-any.whl.metadata
  Downloading requests-2.31.0-py3-none-any.whl.metadata (4.6 kB)
Collecting charset-normalizer<4,>=2 (from requests)
  Obtaining dependency information for charset-normalizer<4,>=2 from https://files.pythonhosted.org/packag
8ac8b1741d5b963797b7c0c869335619e13d4744ca2f67fc11c6fc/charset_normalizer-3.3.2-cp311-cp311-win_amd64.whl.
  Downloading charset_normalizer-3.3.2-cp311-cp311-win_amd64.whl.metadata (34 kB)
Collecting idna<4,>=2.5 (from requests)
  Obtaining dependency information for idna<4,>=2.5 from https://files.pythonhosted.org/packages/c2/e7/a82
68d59ae6a68bf5064484a0718ea5033660af4b54a9/idna-3.6-py3-none-any.whl.metadata
```



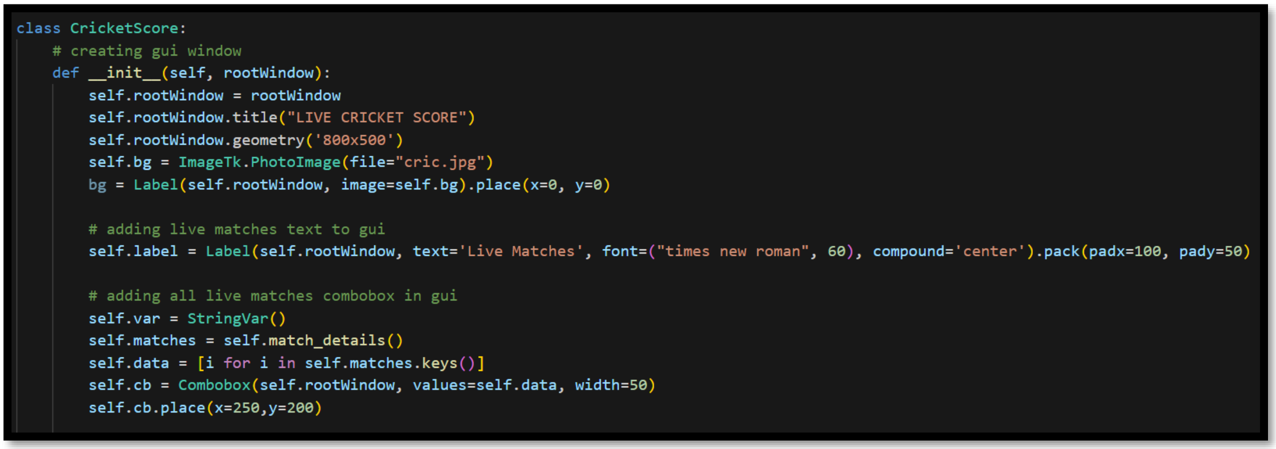
```
PS C:\Users\PMLS\Desktop\CRICSCORE> pip install bs4
Collecting bs4
  Downloading bs4-0.0.1.tar.gz (1.1 kB)
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Preparing metadata (pyproject.toml) ... done
Collecting beautifulsoup4 (from bs4)
  Downloading beautifulsoup4-4.12.2-py3-none-any.whl (142 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 143.0/143.0 kB 160.3 kB/s eta 0:00:00
Collecting soupsieve>1.2 (from beautifulsoup4->bs4)
  Obtaining dependency information for soupsieve>1.2 from https://files.pythonhosted.org/p
```

CODE:

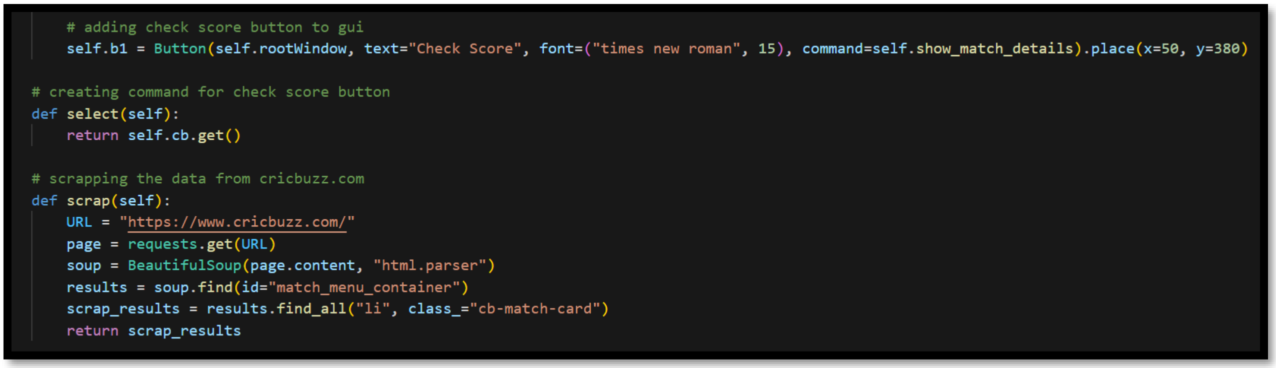


```
from tkinter import *
from PIL import ImageTk
from tkinter.ttk import Combobox
from bs4 import BeautifulSoup
import requests
import re
```

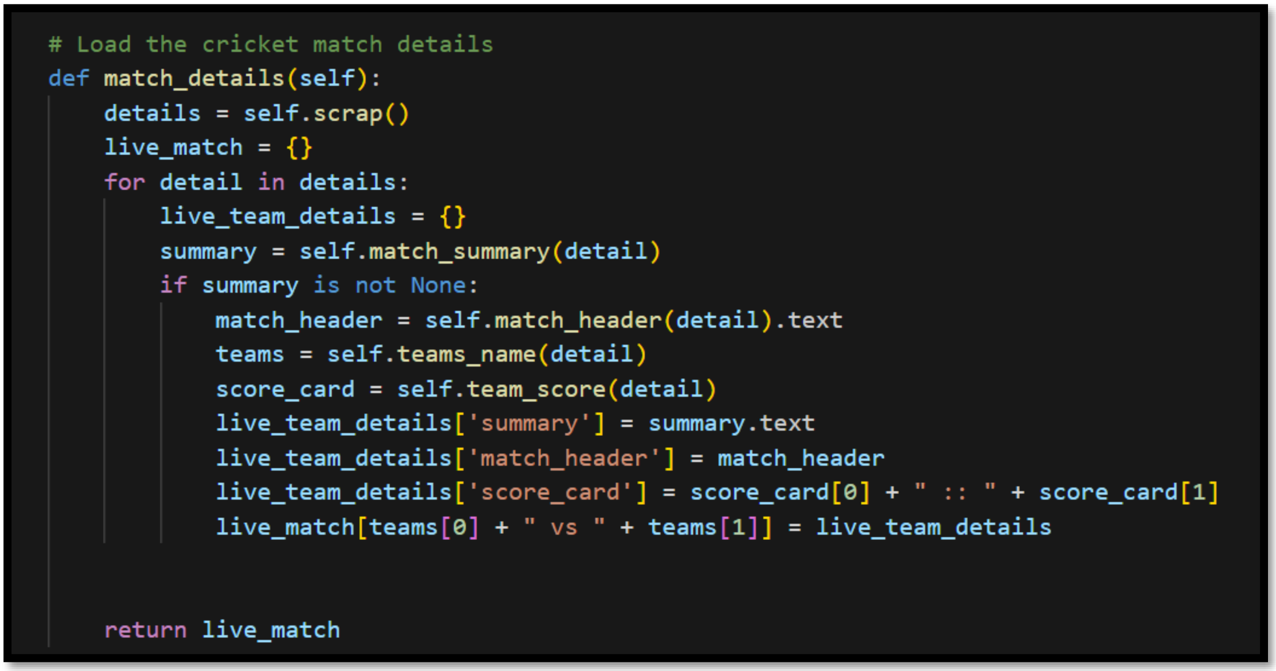
[3] ✓ 0.0s



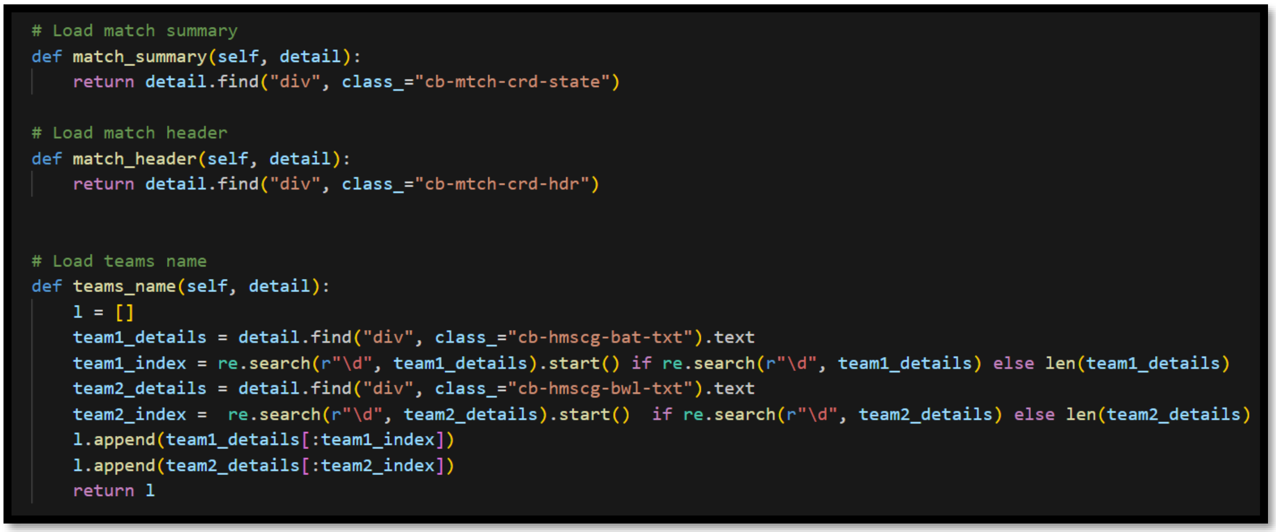
```
class CricketScore:
    # creating gui window
    def __init__(self, rootWindow):
        self.rootWindow = rootWindow
        self.rootWindow.title("LIVE CRICKET SCORE")
        self.rootWindow.geometry('800x500')
        self.bg = ImageTk.PhotoImage(file="cric.jpg")
        bg = Label(self.rootWindow, image=self.bg).place(x=0, y=0)

        # adding live matches text to gui
        self.label = Label(self.rootWindow, text='Live Matches', font=("times new roman", 60), compound='center').pack(padx=100, pady=50)

        # adding all live matches combobox in gui
        self.var = StringVar()
        self.matches = self.match_details()
        self.data = [i for i in self.matches.keys()]
        self.cb = Combobox(self.rootWindow, values=self.data, width=50)
        self.cb.place(x=250,y=200)
```



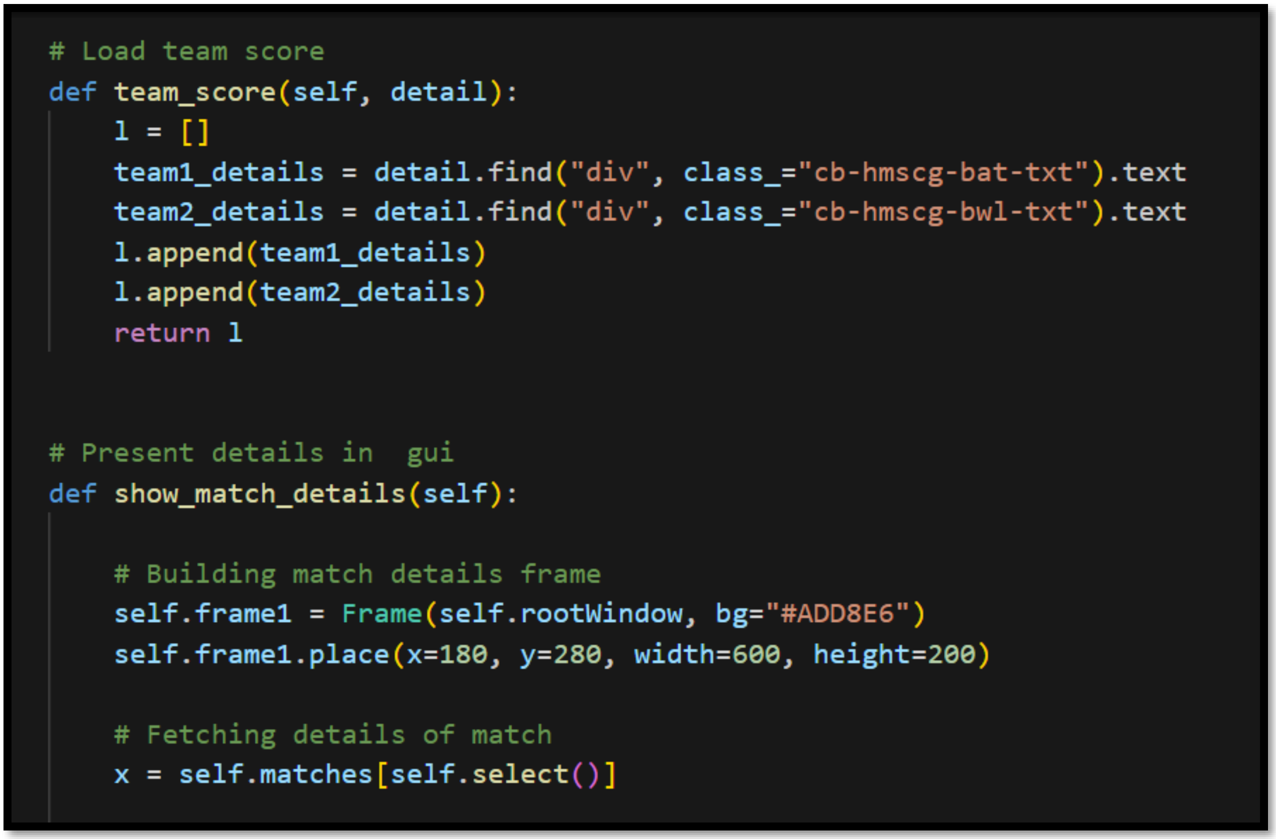
```python
    # adding check score button to gui
    self.b1 = Button(self.rootWindow, text="Check Score", font=("times new roman", 15), command=self.show_match_details).place(x=50, y=380)

# creating command for check score button
def select(self):
    return self.cb.get()

# scrapping the data from cricbuzz.com
def scrap(self):
    URL = "https://www.cricbuzz.com/"
    page = requests.get(URL)
    soup = BeautifulSoup(page.content, "html.parser")
    results = soup.find(id="match_menu_container")
    scrap_results = results.find_all("li", class_="cb-match-card")
    return scrap_results
```



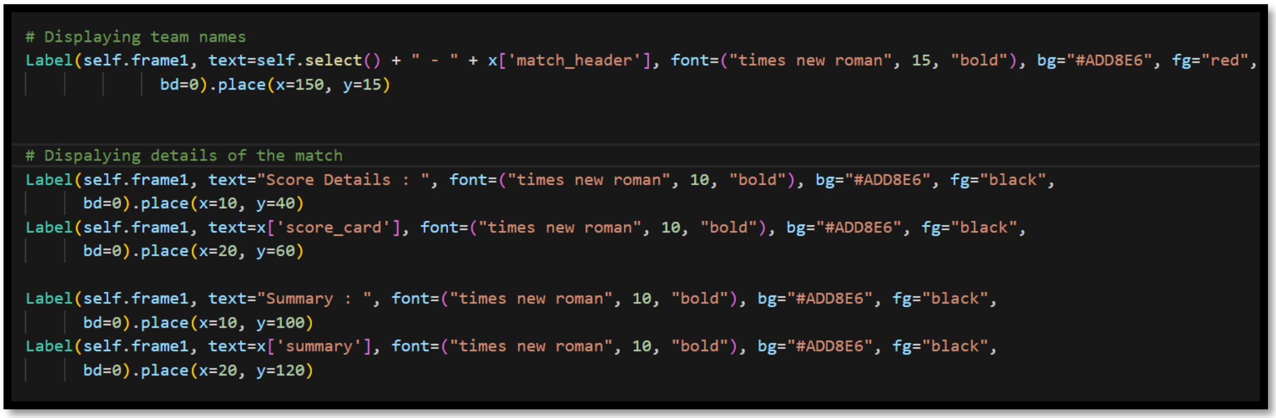
```python
# Load the cricket match details
def match_details(self):
    details = self.scrap()
    live_match = {}
    for detail in details:
        live_team_details = {}
        summary = self.match_summary(detail)
        if summary is not None:
            match_header = self.match_header(detail).text
            teams = self.teams_name(detail)
            score_card = self.team_score(detail)
            live_team_details['summary'] = summary.text
            live_team_details['match_header'] = match_header
            live_team_details['score_card'] = score_card[0] + " :: " + score_card[1]
            live_match[teams[0] + " vs " + teams[1]] = live_team_details


    return live_match
```



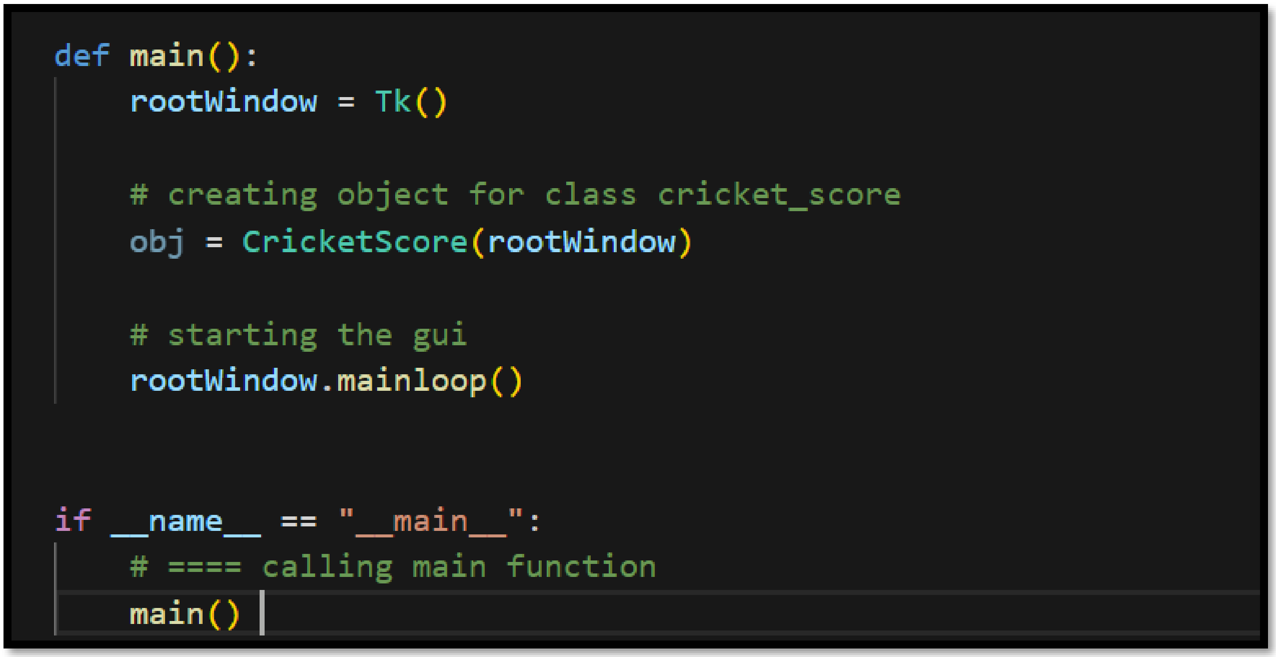
```python
# Load match summary
def match_summary(self, detail):
    return detail.find("div", class_="cb-mtch-crd-state")

# Load match header
def match_header(self, detail):
    return detail.find("div", class_="cb-mtch-crd-hdr")


# Load teams name
def teams_name(self, detail):
    l = []
    team1_details = detail.find("div", class_="cb-hmscg-bat-txt").text
    team1_index = re.search(r"\d", team1_details).start() if re.search(r"\d", team1_details) else len(team1_details)
    team2_details = detail.find("div", class_="cb-hmscg-bwl-txt").text
    team2_index =  re.search(r"\d", team2_details).start()  if re.search(r"\d", team2_details) else len(team2_details)
    l.append(team1_details[:team1_index])
    l.append(team2_details[:team2_index])
    return l
```



```
# Load team score
def team_score(self, detail):
    l = []
    team1_details = detail.find("div", class_="cb-hmscg-bat-txt").text
    team2_details = detail.find("div", class_="cb-hmscg-bwl-txt").text
    l.append(team1_details)
    l.append(team2_details)
    return l


# Present details in  gui
def show_match_details(self):

    # Building match details frame
    self.frame1 = Frame(self.rootWindow, bg="#ADD8E6")
    self.frame1.place(x=180, y=280, width=600, height=200)

    # Fetching details of match
    x = self.matches[self.select()]
```

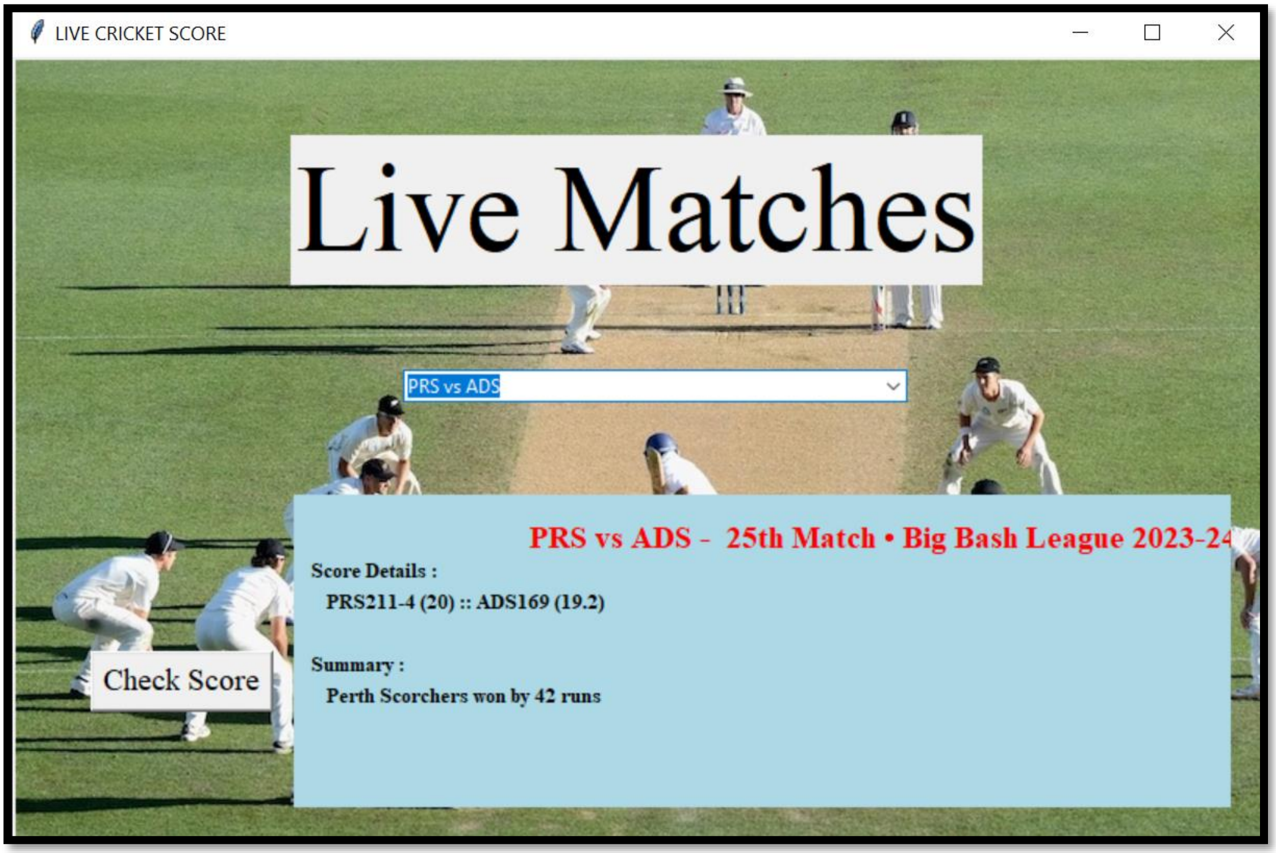


```
# Displaying team names
Label(self.frame1, text=self.select() + " - " + x['match_header'], font=("times new roman", 15, "bold"), bg="#ADD8E6", fg="red",
                bd=0).place(x=150, y=15)


# Dispalying details of the match
Label(self.frame1, text="Score Details : ", font=("times new roman", 10, "bold"), bg="#ADD8E6", fg="black",
        bd=0).place(x=10, y=40)
Label(self.frame1, text=x['score_card'], font=("times new roman", 10, "bold"), bg="#ADD8E6", fg="black",
        bd=0).place(x=20, y=60)

Label(self.frame1, text="Summary : ", font=("times new roman", 10, "bold"), bg="#ADD8E6", fg="black",
        bd=0).place(x=10, y=100)
Label(self.frame1, text=x['summary'], font=("times new roman", 10, "bold"), bg="#ADD8E6", fg="black",
        bd=0).place(x=20, y=120)
```

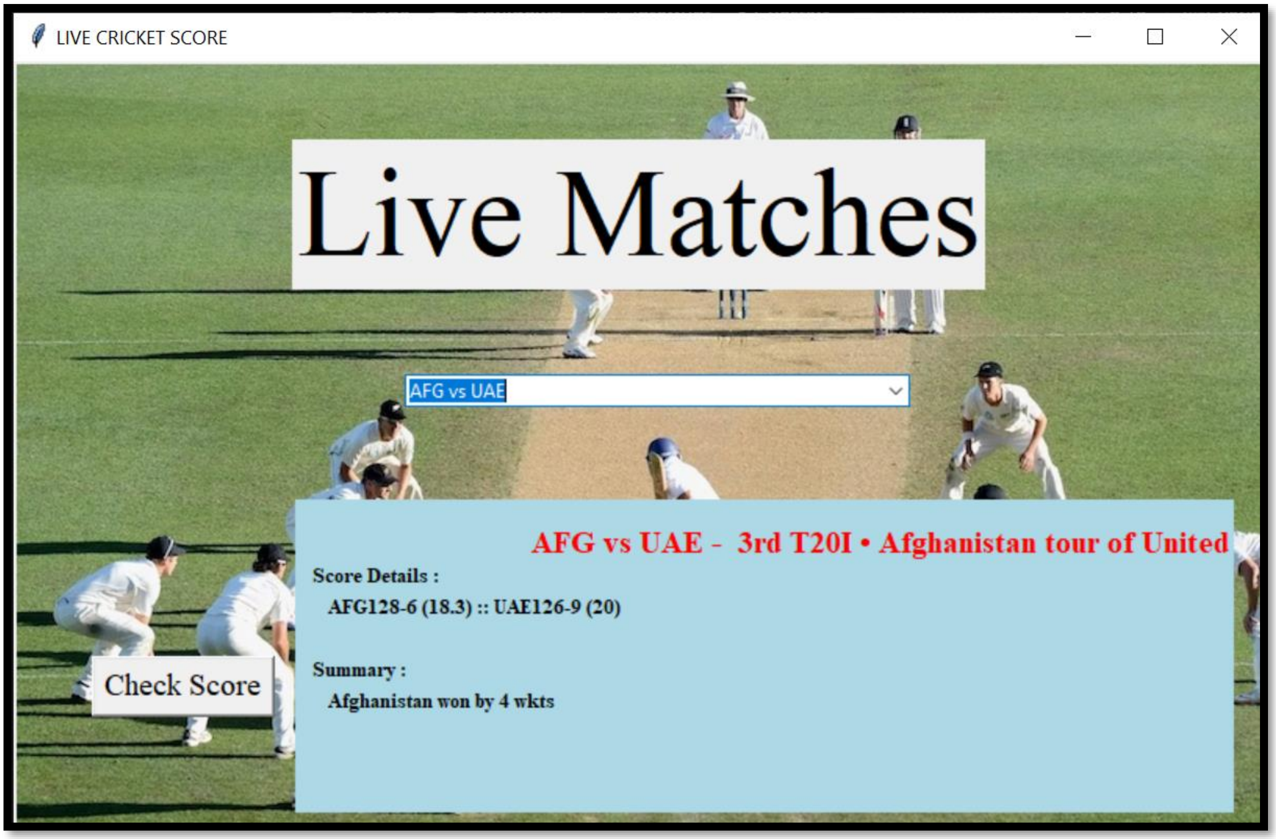


```
def main():
    rootWindow = Tk()

    # creating object for class cricket_score
    obj = CricketScore(rootWindow)

    # starting the gui
    rootWindow.mainloop()


if __name__ == "__main__":
    # ==== calling main function
    main()
```

OUTPUT:



```
PRS vs ADS -  25th Match • Big Bash League 2023-24
Score Details :
  PRS211-4 (20) :: ADS169 (19.2)

Summary :
  Perth Scorchers won by 42 runs
```



AFG vs UAE - 3rd T20I • Afghanistan tour of United

Score Details :
  AFG128-6 (18.3) :: UAE126-9 (20)

Summary :
  Afghanistan won by 4 wkts